CONNEXIONS
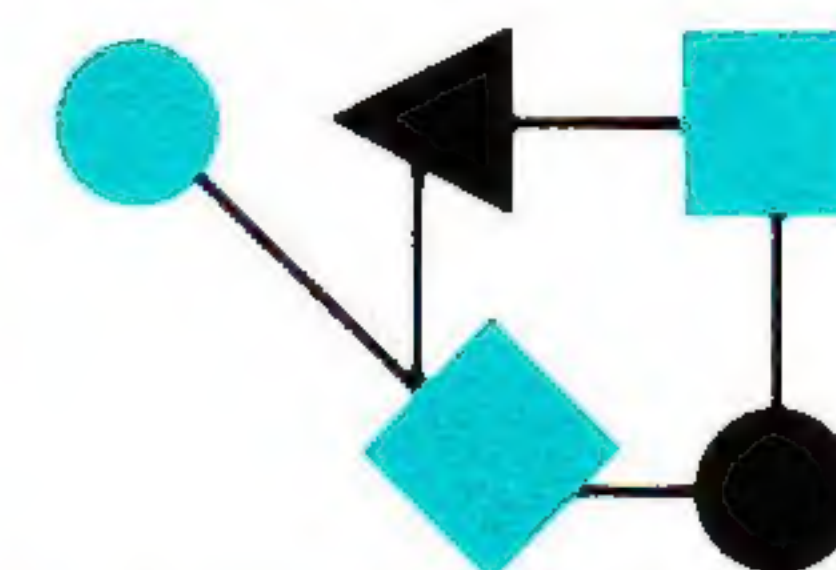
The Interoperability Report

*ConneXions—
The Interoperability Report
tracks current and emerging
standards and technologies
within the computer and
communications industry.*

In this issue:

From the Editor

Belated Happy New Year and welcome to another volume of
ConneXions. As a regular reader of this publication you will no doubt
have noticed a certain irregularity in its delivery. Holidays, travel
schedules, authors, production problems, and the timing of our
NetWorld+Interop events all conspire and may result in occasional
delays. I cannot promise that this situation will ever be completely
eliminated, but I will promise to deliver 12 issues to you each year.

Speaking of promises, I said that I would print another list of Inter-
net books by the end of 1994 and I didn't do so. The reason is simple:
there are now more than 200 books with the word "Internet" in the
title and a new one seems to appear each week. Producing a com-
plete list in print seems like it would chew up too many pages, but I
will investigate the feasibility of compiling such a document for
perusal on our World-Wide Web server **http://www.interop.com**.
On that server you can now find a complete index of *ConneXions*
back issues. A hardcopy version of the cumulative index (1987–1994)
is in preparation and will be mailed to you soon.

As reported in our August and November issues, the current NSF-
NET backbone will eventually be replaced by a system of inter-
connected network providers. In November, Jessica Yu, Enke Chen,
and Laurent Joncheray of Merit Network Inc., described a proposed
routing solution for the initial system of *Network Access Points*
(NAPs). This month's article, by the same authors, is a follow-up
which explains the routing solution in more detail.

In our series "Back to Basics" we look at Internet electronic mail,
probably the most commonly used application. The article is by
David Crocker, an Internet old-timer and author/editor of several
standards documents for e-mail, most notably RFC 822.

Since e-mail is something I use every single day, I have compiled a
list of my personal e-mail dislikes to serve as a starting point for
discussion. Your comments and suggestions are welcome, send them
via e-mail to **connexions@interop.com**

Check out the special double issue of *People* magazine for December
26th 1994! You'll find an entry for Vint Cerf under the heading "The
25 Most Intriguing People of the Year!" The Internet has certainly
reached mainstream notoriety. We'll continue to cover Internet tech-
nology and related systems in *ConneXions*. Stay tuned in 1995.

Routing for the Initial ATM-NAPs In the Presence of the Special ANS/NSFNET Attachment

by
Jessica Yu Enke Chen & Laurent Joncheray,
Merit Network, Inc.

Introduction

During the transition to the new NSFNET architecture, the ANS/ NSFNET will be connected to the three priority *Network Access Points* (NAPs) in Chicago, New York and San Francisco. Some of these NAPs are being implemented using ATM technology, thus they are usually referred to as ATM-NAPs. The initial ATM-NAP architecture includes: PVC-based configuration, DS3 data rate, and use of the protocol stack RFC 1490/AAL5 [2]. As detailed in another article in this journal [1], due to a protocol encapsulation mismatch problem, an intermediate router needs to be introduced in order to connect the *Route Server* (RS) to the ATM-NAP. A routing design was presented in [1] to deal with the special RS connection. That design would be adequate for routing in the ATM-NAP if all *Internet Service Providers* (ISPs) could connect directly to the ATM-NAP. It turns out, however, that the ANS/NSFNET router also requires a special arrangement for its connection to the ATM-NAP due to the interface mismatch problem. This special arrangement in turn requires some extensions to the design presented in [1].

While the design in [1] only deals with the special RS connection, this article includes the extensions that are necessary to handle more than one special attachment (via intermediate routers) to the ATM-NAP. It also discusses the configuration options to allow for the access to the RS from outside the ATM-NAP. This article essentially provides for a complete routing solution for the initial ATM-NAP.

For the sake of brevity, this article does not make any distinction between the concept of an ISP and a *Network Service Provider* (NSP). Also, the term "ISP" is frequently used to refer to an ISP's router that is directly connected to an ATM-NAP (via a ADSU). The ANS/ NSFNET is spelled out explicitly due to its special attachment to the ATM-NAP. It is highly recommended that one be familiar with [1] as this article omits many details that have been discussed in that article.

The planned RS connection

As detailed in our previous article, an intermediate router needs to be used to connect the Route Server (RS) to the ATM-NAP due to an encapsulation mismatch problem. However, to realize its intended functionality, the RS must still be one logical hop away from all the ISPs that are connected to the ATM-NAP, regardless of the presence of the intermediate router. Using the routing design presented in [1] will meet this requisite. The key to the design is to make the intermediate router function like a layer 2 device via careful address assignment and the use of Proxy ARP. The design has shown to be adequate and feasible provided that all ISPs are connected directly to the ATM-NAP without using intermediate routers.

Special ANS/NSFNET attachment

Currently the ANS/NSFNET router cannot accommodate the ATM interface. Thus, for its connection to an ATM-NAP, an intermediate router also needs to be introduced. However, even in the presence of the extra router, it is still intended that the ANS/NSFNET be adjacent to (i.e., one logical hop away from) the other ISPs attached to the ATM-NAP so that routing decisions are not otherwise affected. As a result, a special arrangement, derived from the one designed for the RS connection in [1], has been made.

This arrangement consists of the following components:

- Use an intermediate router (e.g., Cisco) to connect the ANS/NSFNET to the ATM-NAP. More specifically, a border router (ENSS) of the ANS/NSFNET is connected to an FDDI ring, and an intermediate router is connected to both the ATM-NAP and the FDDI ring.

- Assign addresses in such a manner that the intermediate router would perform Proxy ARP for communication between the ENSS and other ISPs. This would make the ENSS one logical hop away from other ISPs.

- Set up external BGP sessions between the ENSS and other ISPs for the purpose of exchanging traffic. (Peer with the RS instead to exchange routing information once the RS is present.)

- Set up an external BGP session between the ENSS and the intermediate router so that the intermediate router would have the same routing knowledge as the ENSS. This is necessary for communication between the networks behind the ANS/NSFNET and other ISPs. Note that the intermediate router requires a separate AS for external peering, but the AS will not be visible to the ISPs.

It is noted that some other options for dealing with the extra router have been evaluated and eliminated from further consideration. One option was to integrate the intermediate router into the routing domain (or AS) of the ANS/NSFNET, but currently this is not possible because of the incompatibility of the *Internal Gateway Protocols* (IGPs), and different levels of supported routing policies in the ANS/NSFNET and the intermediate router. Another option was to treat the intermediate router as a separate AS, and make the AS visible to the ISPs and the Internet. That is, set up an external BGP peer between the ENSS and the intermediate router, and let the intermediate router peer with the ISPs. The concern is that there would be an extra AS number in the AS path, which may impact routing decisions.

RS connection options

The presence of the special ANS/NSFNET attachment makes available several options to connect the RS to the ATM-NAP:

(a) The RS connects to the ATM-NAP via another intermediate router, independent of the ANS/NSFNET arrangement.

(b) The RS connects to the same FDDI ring with the ANS/NSFNET and shares the intermediate router and the ADSU.

(c) The RS connects to the same FDDI ring with the ANS/NSFNET, but has its own ADSU and intermediate router (with static ARP entries configured for the RS).

While Options (b) and (c) would reduce cost and simplify routing (almost no modification is needed for the solution presented in [1]), Option (a) has been chosen because of considerations of network management, traffic load, routing policy and other issues.

With Option (a), however, the communication of the RS and the ENSS can no longer rely on Proxy ARP as in [1] because of the additional intermediate router that connects the ENSS to the ATM-NAP. The following section details the extensions that are necessary to accommodate the special ANS/NSFNET attachment.

3

Routing for the Initial ATM-NAPs *(continued)*

Routing and addressing

The extensions of [1] that are needed to accommodate the special ANS/NSFNET attachment are as follows:

- Configure static routes on the intermediate routers that connect the ENSS and the RS to the ATM-NAP, respectively.

- Assign addresses in such a manner that the intermediate router between the ATM-NAP and the ENSS would also perform Proxy ARP for communication between the ENSS and the RS.

As in [1], the ATM-NAP interfaces, the FDDI ring for the ANS/NSFNET, and the FDDI ring (or Ethernet) for the RS, shall all be configured as different logical subnets. The routing and addressing details are described in the following sections.

Address assignment

The RS, ENSS and other ISPs' routers attached to the NAP shall be configured to be in a *Logical IP Subnet* (LIS). For the intermediate router that connects the ENSS to the ATM-NAP, its interface to the ATM switch shall be configured as in that same LIS but with a longer mask; and its FDDI interface shall be configured to be in a LIS with the ENSS but with a longer mask. For the intermediate router that connects the RS to the ATM-NAP, its interfaces to the ATM switch shall be configured as in the same LIS with other ISP's routers but with a longer mask; and its FDDI (or Ethernet) interface shall be configured to be in a LIS with the RS but with a longer mask.

For example, the address assignment for the San Francisco ATM-NAP provided by Pacific*Bell is shown in Figure 1, in which all addresses have the prefix 198.32.128. Also in Figure 1, the ENSS is a router of the ANS/NSFNET; ISP1 and ISP2 are ISPs' routers that are directly connected to the ATM-NAP. X and Y are networks behind the ENSS and ISP1, respectively. R0 and R1 are the intermediate routers that are used to connect the RS and the ENSS, respectively, to the ATM-NAP. The shaded boxes are ADSUs.
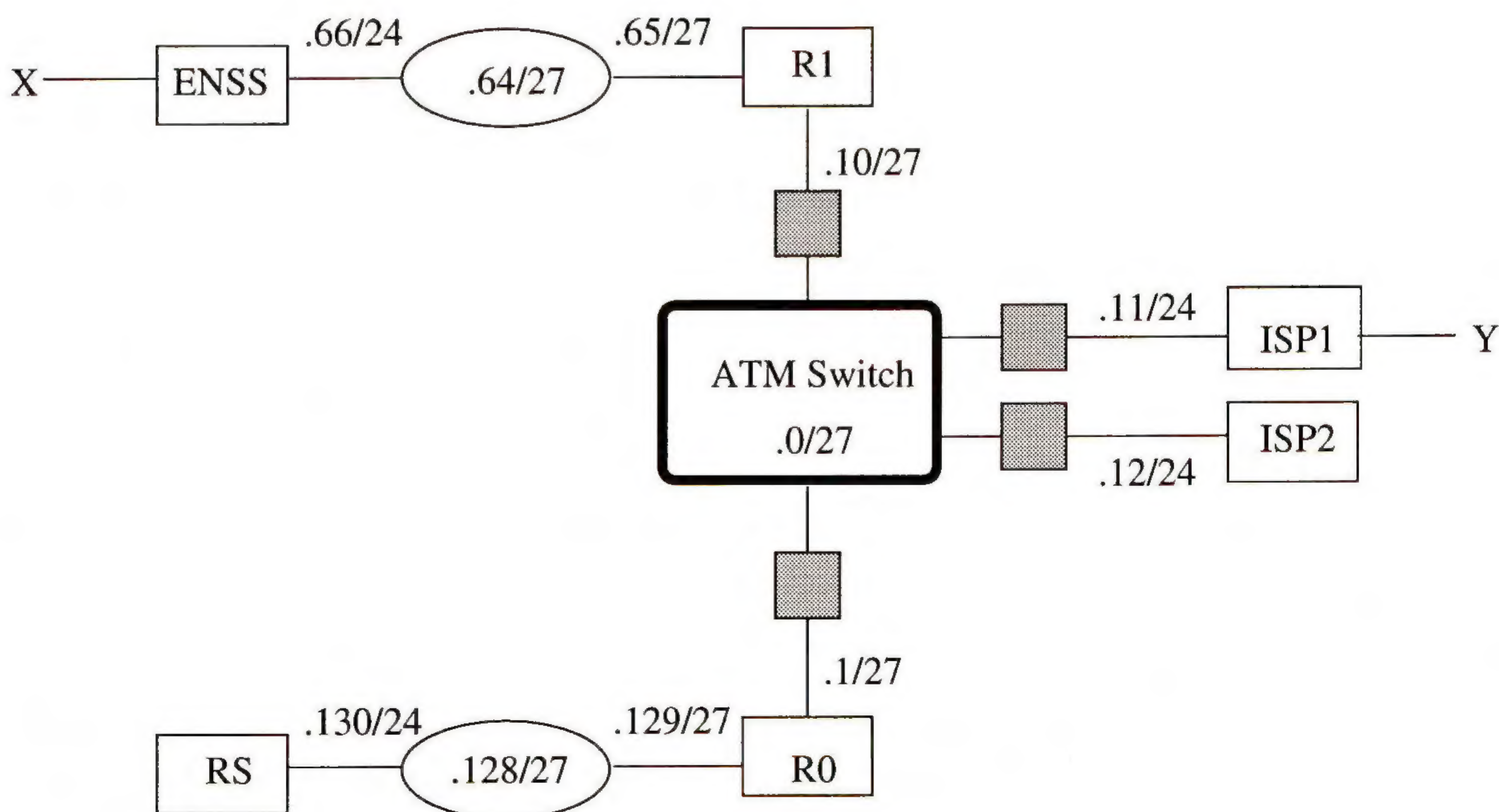

Figure 1: Address Assignment (198.32.128.x)

Routing configuration

The routing configuration involves setting up BGP peering sessions and some static configuration.

- External BGP Peering Sessions:
  - RS (198.32.128.130) with the ENSS and other ISPs
  - ENSS (198.32.128.66) and R1 (198.32.128.65).

- Static Configuration:

Router	Destination	Next Hop
R1	RS (198.32.128.130)	R0 (198.32.128.1)
R0	ENSS (198.32.128.66)	R1 (198.32.128.10)
ISP	RS (198.32.128.130)	VPI/VCI, or DLCI
ISP	ENSS (198.32.128.66)	VPI/VCI, or DLCI

- In this particular case, the static route configuration for R0 and R1 is no worse than a dynamic route configuration as there is only one path between the RS and the ENSS within the ATM-NAP.

- In order for the correct next hop (e.g., 198.32.128.11 for Y) to be accepted on R1, some implementations may require that the ENSS peer with the other interface (198.32.128.65) of R1 instead.

- Depending upon implementations, the static route configuration for the ISPs may also work besides the VPI/VCI or DLCI configuration.

Reachability of the RS

In order to reach the RS from networks beyond the ATM-NAP, the router R0 needs to have routes to these networks. To achieve this, there are a few options depending upon the number of networks that need to reach the RS.

(a) If the number is small and there is only one path from the RS to these networks, then static route configuration on R0 will do.

(b) If the number is large, an external BGP session can be configured between the RS (198.32.128.130) and R0 (198.32.128.129). This will work for all routes except those learned from the ENSS, and they need to be filtered out from RS to R0 to prevent a routing loop. As a remedy, an external BGP session between R0 and R1 can be configured so that R0 learns all networks behind the ANS/NSFNET. This routing setup would require two additional peering sessions (the RS with R0, and R0 with R1).

(c) If the number is large, another approach is to have R0 do external BGP peering with the ENSS and all the other ISPs. But doing so would double the number of peering sessions, which is certainly not desirable.

Option (b) is preferred if the number is large or if there exist multiple paths from the RS to a network that needs to reach the RS.

How does it work?

Given the configuration as presented previously, this section illustrates how the traffic would flow both within (i.e., packets originate and terminate within the ATM-NAP) and beyond the ATM-NAP (i.e., packets originate or terminate outside the ATM-NAP but need to traverse the ATM-NAP). Several trivial cases are not discussed here, including communication between two ISPs within the ATM-NAP, between the RS and a network behind an ISP, and between networks behind two ISPs.

Routing for the Initial ATM-NAPs *(continued)*

Communication within the ATM-NAP

• *Between the RS and an ISP:* The system relies on Proxy ARP and static VPI/VCI, or DLCI configuration. This part is the same as in [1]. When the RS sends a (routing) packet to an ISP, the RS will ARP for the destination. R0 will return a Proxy ARP reply as the other interface of R0 is on the same subnet as the destination. As a result, the packet will be delivered from the RS to R0 and to the destination.

When an ISP sends a packet to the RS, it will use the statically-configured mapping to send the packet to R0. And R0 will deliver the packet to the RS.

• *Between the RS and the ENSS:* The system relies on Proxy ARP and the static route configuration. When the RS sends a packet to the ENSS, the RS will ARP for the destination since the ENSS appears to be on the same subnet with the RS. The router R0 will return a Proxy ARP reply to the RS since R0 has a (static) route to the destination ENSS with R1 as the next hop. As a result, the packet will be delivered from the RS to R0, and to R1, and to the ENSS. The process is similar for packets sent from the ENSS to the RS.

• *Between the ENSS and an ISP:* This part is similar to the communication between the RS and an ISP.

Communication beyond the ATM-NAP

• *Between the RS and a network behind the ENSS:* Both R0 and R1 also have routes to destinations reachable via the ANS/NSFNET because of the peering sessions between the ENSS and R1, and the RS and R0. For example, the RS would have route X with the ENSS (198.32.128.66) as the next hop, and R0 would have route X with R1 as the next hop. And R1 would have route X with the ENSS as the next hop. When the RS sends a packet to X, it will ARP for the ENSS as the ENSS appears to be on the same subnet. R0 will return a Proxy ARP reply because of the (static) route to the ENSS. When R0 receives the packet, it will forward it to R1 (the next hop for X). The packet will be further forwarded to the ENSS by the R1, and to the destination X. For a packet sent from X to the RS, once it is received by the ENSS, the traffic flow will be all within the ATM-NAP, which has been discussed in the previous section.

• *Between the ENSS and an ISP:* R1 would have the correct next hop due to the external BGP peering session between the ENSS and R1.

For example, due to the peering with the RS, the ENSS would have route Y with ISP1 (198.32.128.11) as the next hop; and ISP1 would have route X with ENSS (198.32.128.66) as the next hop. Due to external BGP peering session between the ENSS and R1, R1 would have route X with ENSS (198.32.128.66) as the next hop, and route Y with ISP1 (198.32.128.11) as the next hop. So, a packet from X to Y, would go through the ENSS, R1, and ISP1. And a packet from Y to X, would go through ISP1, R1, and the ENSS.

Conclusion

This article presents extensions to the routing design of [1] in order to accommodate the special ANS/NSFNET attachment, thus provides for a complete routing solution for the initial ATM-NAP. Clearly this solution can easily accommodate a reasonable number of ISPs that may need to use intermediate routers for their connection to the ATM-NAP due to problems of either interface mismatch or protocol encapsulation mismatch.

We are fully aware of the level of detail that the solution involves. The solution is not perfect, but it is certainly preferable to alternatives that have been considered. Furthermore, the solution has proved to be adequate to the immediate problems we are facing for the initial deployment of the ATM-NAPs. It is expected that the routing at the ATM-NAPs will be greatly simplified once more ATM products become available.

Acknowledgements

References

[1] Yu, J., Chen, E. and Joncheray, L., "A Routing Design for the Initial ATM NAP Architecture," *ConneXions*, Volume 8, No. 11, November 1994.

[2] Salo, T.J., "ATM NAP Workshop Conclusions and Summary," Draft, September 1994.

[3] Estrin, D., Postel, J., and Rekhter, Y., "Routing Arbiter Architecture," *ConneXions,* Volume 8, No. 8, August 1994.

[4] Rekhter, Y., and Li, T. (Editors), "A Border Gateway Protocol 4 (BGP-4)," RFC 1654, July 1994.

[5] Plummer, D., "An Ethernet Address Resolution Protocol," RFC 826, November 1982.

[6] Postel, J., "Multi-LAN Address Resolution," RFC 925, October 1984.

JESSICA (JIE YUN) YU received her B.S. in Computer Science from Shanghai Jiao Tong University in China and M.S. in Computer Engineering from The University of Michigan. She is currently a Senior Internet Engineer of Merit Network Inc. and is one of the co-PIs of the Routing Arbiter Award from NSF. She joined Merit in 1987 and was a key contributor in the initial engineering and deployment of the T1 and T3 NSFNET backbone and has since worked on the NSFNET engineering and operation. The major focus of her work has been NSFNET routing architecture design, engineering and implementation, and Internet advanced routing research. She has performed pioneering work on BGP and its deployment and is one of the architects of CIDR. She co-authored several RFCs and co-chairs the IETF CIDR Deployment Working Group. She is currently working on various Routing Arbiter related tasks including Routing Registry design, NAP routing design and ATM-NAP testing. She is also a member of FEPG (Federal Networking Engineering Planning Group). E-mail: `jyy@merit.edu`

ENKE CHEN received the B.S. degree in Instrumentation and Automation from Tsinghua University, Beijing, China, in 1985, and the M.S.E. and Ph.D. degrees in Electrical Engineering: Systems from the University of Michigan, Ann Arbor, in 1987 and 1991, respectively. Mr. Chen joined Merit as an Internet Engineer in 1992. He has worked on the logical modeling and performance analysis of discrete-event dynamic systems, NSFNET routing design and engineering, developing software to study and improve the NSFNET routing and its configuration process, and more recently on ATM-NAP testing and routing design, and other Routing Arbiter related projects. Mr. Chen is a member of IEEE. E-mail: `enke@merit.edu`

LAURENT JONCHERAY received a B.S degree in Electronics from the "Ecole Centrale de Paris" in 1990 and a M.S. degree in Computer Networks from the "Telecom Paris" in 1991. He worked one year at CERN (Switzerland) in the Network Department and joined Merit Network, Inc. in 1993. He has developed tools to automate router configuration generation and built software for network traffic statistic collection and analysis. He is currently an Internet Engineer involved in the design and implementation of various Routing Arbiter tasks. His work focuses on router configuration tools, routing analysis, security and ATM NAP testing. E-mail: `lpj@merit.edu`

Back to Basics:
Internet Electronic Mail
by David Crocker, Brandenburg Consulting

Big bang

When the ARPANET was first built, in 1969, there were grand visions of its possible uses. The only problem was that most of the concrete planning for the network focused on the underlying packet-switching mechanisms and little was devoted to the development of application services. In RFC 1000 [16], Steve Crocker discusses this early work on the upper layers and comments, "We had no official charter. Most of us were graduate students and we expected that a professional crew would show up eventually to take over the problems we were dealing with." But the professional crew never did arrive and the graduate students slogged ahead, developing a few basic protocols, particularly for terminal access and file transfer.

Out of the primordial soup

While electronic mail was a part of the original vision, it was not a part of the main development work. Ray Tomlinson, of Bolt Beranek and Newman (BBN) is credited with modifying BBN's TENEX operating system mail-sending program, *sndmsg*, to support an *ad hoc* protocol for transferring to other TENEXs. Users specified the remote host by appending "@hostname" to the recipient's login name. The *sndmsg* program simply appended the user's message to the end of a recipient's mailbox file. In fact, anyone could append any kind of data. No special privileges were required; when the cross-net facility was added, the server just appended whatever it received. There were no format standards, although the BBN program did create objects with a small set of headers of the same type we are used to seeing today.

Since most of the ARPANET research community used TENEX, the enhancement propagated quickly, creating a functioning e-mail service around the net. But "most" is not the same as "all" and the community already had the view that protocols needed to be independent of any particular platform. During 1971, there were some discussions about a "mailbox" protocol, but it did not reach fruition.

Accidental Tourist

In fact, the e-mail transfer protocol for the next 10 years of the ARPANET—the remainder of its existence—was specified as two, ancillary commands in the *File Transfer Protocol* (FTP) [5]. Better still, it was not specified during the formal working group discussions for creating FTP. Apparently, the editor of the FTP specification was attempting to beat the final draft into acceptable form one Saturday night at MIT, when a co-worker walked by and began discussing e-mail. He suggested adding a couple of commands to FTP to support the transfer function. The editor did just that, without further discussion among the working group. The specification was circulated for review. No one in the community objected; and some implemented it. Thus the ARPANET got its electronic mail protocol.

Besides adding a key function in the ARPANET suite, this demonstrated an elaboration on the group's general style of doing protocol development that largely persists: work is done by a core group and reviewed by a larger group. Individual initiative is often accepted easily, when it causes no special controversy.

The two commands differed in the mechanism they used for sending the message content. FTP, then and now, uses two channels (two transport connections.) One is for commands and the other is for data. The `MAIL` command sent the message content as part of the control channel and the `MLFL` command sent it over a data connection.

To send data over the control channel, the command needed a mechanism for indicating the end of the message content and the convention of having a line containing only a period (`CRLF.CRLF`) was specified. While the mechanism for sending messages via the data channel was implemented and used, the simplicity of the "inline" approach, using the control channel, was more appealing and accounts for carrying this mechanism over to the current e-mail transfer protocol, SMTP.

Both commands allowed specifying only one address at a time. That is, for each recipient, a separate transfer of the entire message was required. (Some implementations allowed referencing multiple recipients in the command line, but this was not widely supported.) While this mechanism caused a bit of inefficiency when sending to several recipients—and especially when sending to a mailing list—it was not felt to be a major burden for some years and was finally remedied by SMTP. Most messages had short distribution lists.

Getting the format right

It is a measure of the ARPANET's informality that there was specification and use of a standard for *sending* e-mail objects several years before there was any concerted effort to define the *format* of the object itself. In part, this was because the original BBN *sndmsg* program generated a simple, reasonable format that most of the community was using. It contained an initial set of headers for specifying author, addressees, subject and creation data; this was followed by a free-form body with lines of text. There was informal documentation of that format, and most other systems conformed to the same, basic conventions. Sometimes, however, messages with remarkably strange formatting would show up in the inbox. (Unlike today, of course...)

The popularity of ARPANET e-mail led, inevitably, to discussions about improving it. As remains true in today's Internet, those with the motivation to talk about the issues were recruited to work on fixing them. This led to RFC 733, the first codification of e-mail object formats [6]. The document pointedly retained the style of e-mail format that was already in use, rather than trying to create a radically new and different system. (There was considerable debate about this philosophical choice, but the decision was essential for protecting the installed base.) Second, the document was the first ARPANET/Internet specification to declare itself a standard. In spite of careful coordination with the ARPA program manager overseeing the effort, the publication of RFC 733 raised quite a few hackles around the net. Conformance to specifications was entirely a matter of individual choice; there was no precedent for being told one had to use a particular specification.

The model of connectivity on the ARPANET was that every host could talk directly to every other host. Since the number of hosts was relatively small, this meant that a single, "flat" name space was adequate. That is, there was one set of host names, maintained in a single, global table. (`HOSTS.TXT`)

The specification used a formal notation, called *Augmented Backus–Naur Form* (ABNF) which was relatively popular among the ARPANET community. With respect to e-mail format, RFC 733 continued existing practice for the basic `name: value` format of headers:

    Subject: e-mail is fun

and the `mailbox@host` format of addresses:

    pogran@mit-multics

9

Internet Electronic Mail *(continued)*

It codified such things as multi-line headers:

```
Subject:some subject lines just need to be longer
        than fits on one line
```

and date formats:

```
Date: Mon, 19 Dec 94 9:30:51 PST
```

The message content, or body, was specified to be simply a series of lines of ASCII text, without structure.

RFC 733 did attempt some innovations. The specification provided for:

- Distinguishing human names from mailbox names:
  (`Dave Crocker <dcrocker@udel-relay>`),

- Distinguishing originator roles: (`From/Sender/Reply-to`),

- Source routing: (`mailbox@host3@host2@host1`),

- Uninterpreted comments:
  (`Paul (DNS guru) Vixie <paul@vix.com>`)

and

- Assorted optional fields, such as `Message-ID`, `Keywords` and `References`,

It also tried to permit a very wide range of addressing formats, including naming for group aliases and to reference postal destinations. User-defined headers were explicitly allowed.

Source-routing was never useful. Given the hands-on, experimental nature of the net, it's surprising that the feature was such a complete failure, however it forced pursuit of an unplanned piece of design cleverness, namely the locally-defined `mailbox` field, affectionately called the "left-hand-side." Only the right-hand-side `hostname` was expected to be interpretable globally. Hence, sites could define any extended interpretation of the left-hand, local side that they wished. This led to Unix UUCP use of additional source-routing, with exclamation (bang) signs:

```
uwisc!lhl@udel-relay
```

and the author's own extension for CSNet, with percent sign used to subdivide the left-hand side for one addition e-mail "hop":

```
lhl%uwisc@udel-relay
```

[The per-cent hack was intended only as a short-term mechanism until Domain Names were fully operational. The persistence of the mechanism into today's Internet shows either that the evil men do really does live after them, or that it was a brilliant invention. Unfortunately, the author believes that the former is the more likely truth.]

Of the innovative work, the `Sender/From/Reply-to` distinction seems to have been the most successful. It distinguished the author(s) of the message, in the `From` field, from the person posting the message, in the `Sender` field. It also let the originator specify that replies were to go to yet-another address. The `From/Reply-to` distinction is in heavy use today; however it is rare for the `Sender` field to be different from `From`.

These fields were RFC 733's one major enhancement that pertained to user-to-user interaction, rather than user-to-mail system. In general, ARPANET and Internet e-mail lack much discussion or mechanism for user-to-user protocols.

Getting serious about scaling

By the time of the 1983 transition from ARPANET to Internet, the major problem with e-mail was its success. Traffic had increased considerably. Mailing lists were becoming popular, and the inefficiencies of the FTP-based mail transfer commands were noticeably irritating. The number of hosts on the net became large enough that maintaining a single, centralized table mapping their names to their addresses was not feasible. Also, the lessons from USENET and CSNet were that not all e-mail hosts were attached directly to the net, yet there were no integrated methods of addressing these "detached" hosts, other than the "left-hand-side" hacks. Also, then as now, conformance to the e-mail specifications was highly problematic; some effort was needed to adjust the specs to match real-world behaviors.

In other words, ARPANET/Internet e-mail needed further enhancement. The effort took place along 3 lines, each involving the usual style of having a core person doing the leadership and writing, with a collection of discussants providing ideas and feedback. The first line of development was the *Simple Mail Transfer Protocol* (SMTP) [15], the second was an acronym-free enhancement to the format specifications in RFC 822 [7]. [The author occasionally jokes that his major contribution to the later MIME development was to convince the MIME authors to make sure that the specification could be expressed as an acronym, independent of its publication identifier. This makes reference to the specification easier and more stable.] The third created a mechanism for distributed administration and query of the host table [11, 12]. This last activity, called the *Domain Name System* (DNS), was not specific to e-mail activity and discussion of it is left for a separate article.

SMTP

Prior to the development of SMTP, there were several other efforts at creating e-mail-specific transport protocols. Besides the USENET and CSNet work, SMTP's author, Jon Postel, earlier led in an effort to develop specifications for support of multi-media mail [14, 18]. Interest was slight, although the effort did have some influence on the later development of X.400. The later work on SMTP was assiduous in its efforts to keeps things simple. For message submission, it only provided a means of specifying:

- Who the message was from,
- An address list of who the message was to, and
- The message.

Protocol replies are in the usual form, with a 3-digit status code, followed by free-form text. This produces message posting sequences of the style as shown in Figure 1 on the next page. The example is taken from the SMTP specification and shows the dialog between the 'S'ending client and the 'R'eceiving server.

Transfer of the message, itself, is done "inline" rather than through a separate data channel. There is also some provision for basic tracing information, to be added to the message.

SMTP provides commands for verifying (VRFY) the validity of a particular address and for expanding (EXPN) an address list reference into its constituent addresses. Both of these functions are useful, but they require a) direct Internet access for the client, and b) a server with direct access to the full address and list details.

11

Internet Electronic Mail *(continued)*

```
R: 220 BBN-UNIX.ARPA Simple Mail Transfer Service Ready
S: HELO USC-ISIF.ARPA
R: 250 BBN-UNIX.ARPA
S: MAIL FROM:<Smith@USC-ISIF.ARPA>
R: 250 OK
S: RCPT TO:<Jones@BBN-UNIX.ARPA>
R: 250 OK
S: RCPT TO:<Green@BBN-UNIX.ARPA>
R: 550 No such user here
S: RCPT TO:<Brown@BBN-UNIX.ARPA>
R: 250 OK
S: DATA
R: 354 Start mail input; end with <CRLF>.<CRLF>
S: Blah blah blah...
S: ...etc. etc. etc.
S: .
R: 250 OK
S: QUIT
R: 221 BBN-UNIX.ARPA Service closing transmission channe
```

Figure 1: SMTP dialog

RFC 822

Format specifications in RFC 822 sought to maintain the working portions of the earlier RFC 733, clean up assorted minor problems, and add only limited enhancements. These included:

- Structured host domain names (`relay.udel.edu`)
- Tracing information supplied in the `Received` header,
- User-level message re-sending onto another recipient (`Resent-to/Resent-From/Resent-Reply-to`),
- Message body privacy (`encrypted`), and
- The `Postmaster` reserved mailbox address.

Use of host domain names was specified before the Domain Name System was actually developed, and specification of the `Received` header was redundant with the specification in SMTP. In both cases, some inconsistencies between specifications has resulted in awkwardness of service that had to be resolved in the later Host Requirements work [4]. Given the current interest in privacy, it's interesting that the RFC 822 mechanism for encryption was not exploited. On the other hand, the simple provision for a standard `Postmaster` address at every site has been quite helpful for e-mail operations. It means that problems can be reported, without having to know any of the staff at the remote site.

A matter of models

Electronic mail architecture discussions refer to two, primary components: *User Agent* (UA) and *Message Transfer Agent* (MTA). While many in the community may debate the issue, ARPANET/Internet e-mail does conform to this model and was even used as one of the examples during formulation of the model.

The model is quite simple. Is says that one part of the system represents individual users and the other represents the conveyance service. What causes confusion, for some, is that there are implementations which do not provide such a clean distinction, such as by having the conveyance software fill in the `From` and `Date` fields. Recent enhancements to the model are attempting to deal with the reality of gateways, that is, modules which are used to connect together e-mail services which are fundamentally different. This is a topic which remains problematic.

Getting serious about scaling, again

For nearly ten years, no further work was done to enhance Internet e-mail. During that time, however, the Internet grew enormously; use of mailing lists grew enormously; and interconnections through e-mail gateways grew enormously. All of this stressed the service, once again. By the beginning of 1991, increased internationalization of Internet use finally forced the community to pursue some changes. Initially, the focus was strictly upon the requirement to permit carriage of documents containing text in non-English character sets. Two technical views dominated the early discussions. One view was that the e-mail object (RFC 822) needed to be enhanced for labeling of non-ASCII data. The other view was that SMTP needed to be enhanced, for carriage of 8-bit data. Separate working groups were formed; the first produced MIME and the second produced ESMTP. For a thorough introduction to the technical aspects of modern Internet mail service, see [17].

MIME

Multi-purpose Internet Mail Extensions (MIME) [1, 2, 3, 13] expanded its original agenda, just a bit. It developed into a general method for structuring and labeling content. The structuring is permitted to be an arbitrary, nested tree of sub-components. The labeling is for a wide range of data, including different types of text, but also including graphics, audio, and so forth. This opened the doors for general, 8-bit data; however the existing e-mail transport services tended (and still tend) to expect simple lines of 7-bit data, so MIME also provides a mechanism for encoding the data to safely traverse these limited environments. In particular, MIME does not require changes to any of the existing transport infrastructure; it only requires changes to the participating end-user systems.

MIME is really specified as independent of RFC 822. The latter focuses on message headers and the former specifies message content, although it does provide for some additional headers, specific to MIME usage. A side-effect of this independence is that MIME can easily be used in non-e-mail environments, as is already happening for the *World-Wide Web*.

Structuring is accomplished through the use of boundary strings:

```
From:
To:
Subject:
Mime-version: 1.0
Content-type: multipart/mixed; boundary=unique1;

--unique1

This is some top-level, introductory text

--unique1

Content-type: multipart/mixed; boundary=unique2;

--unique2

Content-type: text/plain; charset=ISO-8859-1

This is some nested text using an alternate character
set

--unique2

This text is part of the nested set, but is in the
default, ASCII character set

--unique2--
--unique1
```

13

Internet Electronic Mail *(continued)*

The relationship of the data in the above message is:

```
Headers
top-level text
        first nested text
        second nested text
```

Types of content

The MIME header `Content-type` is used to specify the nature of the contained data. Types are specified in two parts, with the first indicating general category and the second (the sub-type) specifying the exact detail. The MIME specification provided for a basic set of sub-types, in each category, with provision for private conventions and for publicly registering extensions:

`text` is for prose. The parameter `charset=` allows specification of other character sets, for use with non-English languages. A basic set of alternatives is specified in the original MIME document, but more importantly, it can be expanded through separate registration of additional sets.

`multipart` is for basic structuring of MIME data; it allows a collection of parts to be "stapled" together. An innovative form is `multipart/alternative` which indicates that the different parts are equivalent and that the recipient should choose one among them; an example of this function would be to send a document which is in its original word processor form, a simple text form, and a *PostScript* form.

`message` is primarily for data which is an e-mail message, principally in RFC 822 format. However, two, innovative forms are for sending long messages and for retrieving data remotely. The first, `message/partial` indicates that this is one part of the message; hence, a sender may break a long message into parts for the receiver to reassemble. The second, `message/external-body` allows reference to files that are elsewhere and permits specification of various retrieval techniques, such as through e-mail request or through Internet FTP.

`audio` is for sounds; the sub-type `audio/basic` provides initial capability.

`image` is for graphics and pictures; the sub-types `image/gif` and `image/jpeg` provide initial capability.

`video` is for motion pictures; the sub-type `video/mpeg` provides initial capability.

`application` is for data which does not fit into any of the above categories; one example is `application/postscript.`

Binary data in non-binary environments

In order to squeeze the binary objects through limited e-mail pipes that often support only 7-bit data, MIME's `Content-Transfer-Encoding` mechanism allows specification of the "transfer" form of the data. The transfer form is independent of the "representation" form. The former specifies what is necessary to get a bundle of bits through a transport pipe. The latter specifies the canonical, interoperable, semantics-related nature of the data. For example, the basic audio form used in MIME encodes data as true binary. To get it through the usual Internet e-mail service, it would be processed into a line-formatted, hexadecimal, textual form, as described below.

MIME permits the following transfer encodings:

7-bit is the default and the most heavily used; it means that the data naturally conform to a 7-bit, line-oriented environment; it is used for ASCII text data.

8-bit indicates that the data conform to line-orientation, but that the high-bit might be on for some bytes; it is used for non-ASCII text data when the transport allows the high bit to be on.

quoted-printable says that the transmitted form of the data conform to the 7-bit, line-oriented constraints but that the conformance was achieved by specially encoding some of the data; it is primarily intended for text data which has the 8th bit on, but which cannot be passed through the e-mail transport transparently.

base64 indicates that the transmitted form of the data conform to the 7-bit, line-oriented constraints, but that the conformance was achieved by specially coding all of the data into a hexadecimal representation; it is used for true, binary data.

MIME originally included a pure **binary** encoding, but it has not yet received enough implementation and testing experience to be retained. It appears that use of MIME for the *World-Wide Web* will provide the necessary pressure to further develop **binary**.

ESMTP

SMTP extensions were the result of the effort parallel to MIME, to enhance the basic e-mail transport service [8, 9, 10]. It provides for incremental specification of enhancements to SMTP systems, through a registration and announcement mechanism, so that a client SMTP system can ask the server what extensions are supported, before trying to use any of them.

The first extensions specified for this mechanism were for support of MIME **Content-transfer-encoding: 8-bit**, and for determining the maximum size of message that the server will accept. From RFC 1651, an example of an extended SMTP start of session, in which the server supports a range of options could be:

```
S: <wait for connection on TCP port 25>
C: <open connection to server>
S: 220 dbc.mtview.ca.us SMTP service ready
C: EHLO ymir.claremont.edu
S: 250-dbc.mtview.ca.us says hello
S: 250-EXPN
S: 250-HELP
S: 250-8BITMIME
S: 250-XONE
S: 250 XVRB
...
```

Note that the **EHLO** query from the client is a different command than the **HELO** query shown in the earlier SMTP example. The enhancement was designed to save from adding even one extra round-trip transaction across the Internet. Experience has shown that this can be a significant concern, particularly when the Internet gets congested.

The extension mechanism is intended to support the gradual upgrade of the service, rather than for supporting "optional" use of enhancements. That is, Internet services generally do not "negotiate" among a range of services that client and server might choose to implement.

15

Internet Electronic Mail *(continued)*

Mechanisms like the one added to SMTP and MIME serve to permit graceful adoption of new features, while still supporting a core set of useful functionality. The freedom created by such a mechanism is that new features can be specified, tested, and deployed incrementally and independently. In fact, various new features are getting added to MIME and SMTP regularly.

The future

The e-mail industry has been extremely active, recently. It is no longer novel to have an e-mail account. Internet e-mail has grown as explosively as the rest of the Internet, particularly as witnessed by the frequency with which radio, TV, advertising, and business cards cite an electronic mail address along with a telephone number.

With the addition of MIME and ESMTP, Internet e-mail begins to provide a service base that is competitive with other e-mail technologies. Some holes remain, such as a fully deployed security service, but there is considerable e-mail service enhancement taking place in the Internet standards forum (IETF) and in the marketplace. It appears that every e-mail vendor will be offering MIME support over the course of 1995, making fully interoperable, open-systems, multimedia e-mail a realistic goal for organizations.

Acknowledgements

The author is indebted to Ken Pogran, John Vittal, Einar Stefferud and Vint Cerf for participating in the spontaneous review of early e-mail history that we conducted during a "Fireside Chat" at the Boston, 1994 *E-Mail World* conference.

References

[1] N. Borenstein & N. Freed, "MIME (Multipurpose Internet Mail Extensions): Mechanisms for Specifying and Describing the Format of Internet Message Bodies," RFC 1341, June 1992.

[2] N. Borenstein, "A User Agent Configuration Mechanism for Multimedia Mail Format Information," RFC 1343, June 1992.

[3] N. Borenstein, "Implications of MIME for Internet Mail Gateways," RFC 1344, June 1992.

[4] R. T. Braden, "Requirements for Internet hosts—application and support," RFC 1123, October 1989.

[5] A. K. Bhushan, "File Transfer Protocol," RFC 354, July 1972.

[6] D. Crocker, J. Vittal, K. T. Pogran & D. A. Henderson, "Standard for the format of ARPA network text messages," RFC 733, November 1977.

[7] D. Crocker, "Standard for the format of ARPA Internet text messages," RFC 822, August 1982.

[8] J. Klensin, N. Freed, M. Rose, E. Stefferud & D. Crocker, "SMTP Service Extensions," RFC 1651, July 1994.

[9] Klensin, N. Freed, M. Rose, E. Stefferud & D. Crocker, "SMTP Service Extension for 8bit-MIME transport," RFC 1652, July 1994.

[10] J. Klensin, N. Freed & K. Moore, "SMTP Service Extension for Message Size Declaration," RFC 1653, July 1994.

[11] P. V. Mockapetris, "Domain names: Concepts and facilities," RFC 882, November 1983.

[12] P. V. Mockapetris, "Domain names: Implementation specification," RFC 883, November 1983.

[13] K. Moore, "Representation of Non-ASCII Text in Internet Message Headers," RFC 1342, June 1992.

[14] J. Postel, "Structured format for transmission of multi-media documents," RFC 767, August 1980.

[15] J. Postel, "Simple Mail Transfer Protocol," RFC 821, August 1982.

[16] J. K. Reynolds & J. Postel, "Request For Comments reference guide," RFC 1000, August 1987.

[17] M. Rose, *The Internet Message: Closing the Book with Electronic Mail,* Prentice-Hall, 1993.

[18] S. Sluizer & J. Postel, "Mail Transfer Protocol," RFC 772, September 1980.

[19] G. Vaudreuil, "MIME: Multi-Media, Multi-Lingual Extensions for RFC 822 Based Electronic Mail," *ConneXions,* Volume 6, No. 9, September 1992.

[20] N. S. Borenstein, "*Metamail*—Multimedia Mail for the Masses," *ConneXions,* Volume 6, No. 3, March 1992.

[21] G. Vaudreuil, "Networked Voice Messaging," *ConneXions,* Volume 8, No. 11, November 1994.

[22] D. Blum, "An Analysis and Comparison of Internet and X.400 Messaging," *ConneXions,* Volume 6, No. 9, September 1992.

[23] *ConneXions,* Special Issue: Electronic Messaging and Directory Service, Volume 6, No. 9, September 1992.

[24] Stallings, W., "Pretty Good Privacy (PGP)," *ConneXions,* Volume 8, No. 12, December 1994.

[25] Stallings, W., "Cryptographic Algorithms, Part I: Conventional Cryptography," *ConneXions,* Volume 8, No. 9, September 1994.

[26] Stallings, W., "Cryptographic Algorithms, Part II: Public-Key Encryption and Secure Hash Functions," *ConneXions,* Volume 8, No. 10, October 1994.

DAVID H. CROCKER has participated in the development of internetworking capabilities since 1972, first as part of the ARPANET research community and more recently in the commercial sector. He wrote the current Internet standard for electronic mail header formats and was a director and principal architect for MCI Mail. He has worked at a number of Silicon Valley companies, producing TCP/IP, OSI, and network management products and has participated in the development of many Internet standards. Mr. Crocker is a principal with Brandenburg Consulting, specializing in business, marketing and technical planning and design for distributed information products and services. His current Internet technical involvement is with electronic commerce, messaging, and enhancements to Internet transport. E-mail: dcrocker@mordor.stanford.edu

Electronic Mail—A Powertool

by Ole Jacobsen, Interop Company

Introduction

There is little doubt that electronic messaging represents one of largest applications of computer networks. Every day, millions of messages traverse our network. In this article, I will offer some observations on the use and misuse of electronic mail. My perspective is that of a long-time Internet user, your own experience may be quite different, and I would invite anyone to write (send e-mail of course) with comments.

To me, electronic mail is a powertool, full of capabilities, and full of dangers. Like a powertool, it should be used responsibly with the recognition that it can do harm to its user or any bystander if not operated properly.

I've been using electronic mail since 1976. I have seen mail systems (user agents) come and go, subscribed to many mailing lists and USENET newsgroups, and interacted with thousands of people all around the globe via electronic mail. I continue to receive almost all articles in this journal via electronic mail (including "pictures" in the form of, sometimes amazingly large, *PostScript* files).

The beauty of e-mail is that it is so quick and easy to use, and can be as formal or informal as the user wants it to be. This ease of use, however can also create problems.

My list

What follows is a short list of my personal e-mail dislikes. They range from problems with a particular user agent or message transfer system, to human "e-mail behavior." Often, a particular problem is a combination of both bad technology and bad habits. In some systems, it is simply too easy to do the "wrong thing"—it may even be how that system is set up to behave by default.

• *Reply-Sender versus Reply-All:* You've seen this many times, I'm sure. Someone sends a message to a large mailing list you're on: "If you'd like more information...send me a message." The next day, you get 10 messages saying "I'd like more information..." Instead of sending their request to the original poster, they have copied the entire list and in the process managed to annoy all of the readers of that particular list. Such behavior isn't necessarily intentional, however. Some user agents simply have "Reply-All" as the default behavior, and this is not at all obvious to the new user. Also, on a few occasions, a sender may change the `Reply-to:` field of a message. If you're not careful, your reply may again go to a large list.

• *Subscribe and unsubscribe:* The Internet convention is to direct all messages of the form "please add me to this list" to the `-request` address, for instance `ietf-request@cnri.reston.va.us`, rather than to the list itself. This convention is either not very well known, or too often ignored, at least judging by the amount of "administrivia" on most lists. Educating new users about it is a major challenge.

• *Flaming:* You'll rarely see verbal arguments in "real life" that are as intense as those often seen on the net. Somehow, the electronic medium seems to encourage this kind of behavior. While flaming can sometimes be quite entertaining to watch, it also drains a great deal of energy from all involved parties. One form of flaming is what I call "two in the ring." This is when two individuals argue publicly with one another on a large mailing list. But flaming is also common in person-to-person messaging, and can be quite destructive. Settling an argument via e-mail is rarely the best option.

- *Inclusion of the message you are replying to:* My personal preference is to let the `Subject:` field speak for itself, and perhaps augment this with a few short extracts from the message I am replying to when absolutely necessary. Some users—and perhaps more importantly: some user agents—insist on *always* including the *entire* message, usually in front of the reply. In addition to chewing up more network bandwidth, this causes great grief to those of us who receive our mail on portable devices such as alphanumeric pagers. On the other hand, I know several users who swear by this method, it's "...the only way you can keep track of the message you're replying to." (Well, I said this was a personal list of gripes, didn't I?)

- *"New" e-mail systems:* A particular cause of frustration is the increasing number of PC LAN based e-mail systems that are now being gatewayed into the Internet. Often these systems work fine in their original environment, but seldom do they "play by the rules" in the outside world. Take for example a popular e-mail program used for Macintosh systems. If you want to send a message to a large group of people (such as the entire company), you create a group and use an alias in the `To:` field. However, the system has no concept of a real mailing list, and will actually expand the alias and substitute for it every corresponding user name before sending the message into SMTP land. The result is two or three screens full of `To:` addresses. One solution is to instruct users to place the alias in the `Bcc:` (Blind-Carbon-Copy) field. This, however, brings with it the drawback that you no longer can tell who received the message.

LAN-based e-mail systems also often include the concept of an "enclosure" that is, the ability to send along a rich-text document or graphic in addition to plain ASCII text. This is of course a nice feature, but it assumes that all recipients have similar equipment and software. When such systems are gatewayed to the Internet, the result is often files in "BinHex" or "UUENCODE" being sent out to bewildered recipients.

- *MIME:* I suppose I'm slow at adapting to change, nonetheless while MIME is undoubtedly a great enhancement to Internet e-mail, it also introduces some annoying features such as `quoted-printable` which cause simple characters such as carriage-returns to show up as equal signs (=) in plain text messages, if you're not using a MIME compliant mail reader. Marshall Rose tells me that "you cannot have new functionality and backwards compatibility at the same time," but I remain unconvinced.

- *Long lines:* With the advent of workstations, users have the ability to set the size of their terminal "window" to anything they wish. This often leads to messages with lines that are longer than 80 characters. Depending on the terminal of the recipient, the resulting messages are "ugly" (wrapped) at best, and unreadable in worse cases.

- *Long messages:* I suppose one could argue about what constitutes "long," but messages longer than 500,000 bytes are certain to impact the recipient in one way or another. I often get *PostScript* files that are that large, but that's because I ask for diagrams to be sent to me in that fashion. Unsolicited large messages can be a real hassle, particularly when they are sent to mailing lists.

- *Messages you can't reply to:* It happens all too often that I receive a message which I cannot reply to because the `From:` field is an invalid address, lacking the fully-qualified domain name, or is otherwise not to spec.

19

E-mail—A Powertool *(continued)*

• *Stupid gateways:* A number of commercial e-mail gateways strip or mangle important information as messages pass through them. This topic is large enough for an entire article and I know just the person to write it, Stef are you listening?

• *Forwarding loops:* If you have a forwarding mechanism of any kind there is always the potential for disaster. For example, a copy of every message sent to me (with "ole" in the `To:` or `Cc:` field) is forwarded to my radio pager. Now suppose for some reason the mail system on the pager side rejects the forwarded message (this has happened on a couple of occasions). The message, along with some trace and error information, will be returned back to my main mail system where it will promptly be forwarded once more, and so on and so on. Last time this happened, it took only about four hours for the inbound mail file `/usr/spool/mail/ole` on the Unix host I use to grow to 116 Megabytes. However, the looping condition is preventable with the use of smart filters which don't forward messages from `Mailer-Daemon` or `Postmaster.`

Conclusion

With a little thought and consideration, most of these problems can be fixed, either by changing behavior on the part of the users or by improving our e-mail tools. I'm sure that all the technical problems are fairly simple to address, but changing human behavior is a much greater challenge. At the risk of sounding negative, I leave you with a quote from Gene Spafford of Purdue in his "Farewell to USENET" message:

> *People rail about their "rights" without understanding that every right carries responsibilities that need to be observed too, not least of which is to respect others' rights as you would have them respect your own. Reason, etiquette, accountability, and compromise are strangers in far too many newsgroups these days.*

References

[1] D. Crocker, "Standard for the format of ARPA Internet text messages," RFC 822, August 1982.

[2] D. Crocker, "Back to Basics: Internet Electronic Mail," *ConneXions,* Volume 9, No. 1, January 1995.

[3] J. Postel, "Simple Mail Transfer Protocol," RFC 821, August 1982.

[4] M. Rose, *The Internet Message: Closing the Book with Electronic Mail,* Prentice-Hall, 1993.

[5] G. Vaudreuil, "MIME: Multi-Media, Multi-Lingual Extensions for RFC 822 Based Electronic Mail," *ConneXions,* Volume 6, No. 9, September 1992.

[6] *ConneXions,* Special Issue: Electronic Messaging and Directory Service, Volume 6, No. 9, September 1992.

OLE–JØRGEN JACOBSEN has been an active Internet user since 1976 when he went to work for the Norwegian Defence Research Establishment, an early ARPA-NET site. He joined Dan Lynch as employee number three of the newly formed Interop Company (then called "Advanced Computing Environments") in February 1987, just weeks before the very first Interop conference. Since that time, Ole has remained the Editor (and since 1991 Publisher) of *ConneXions—The Interoperability Report*, a monthly technical journal in the field of computer-communications. He is also the Program Director for the NetWorld+Interop conference and tutorials. Ole is a member of the American Guild of Organists (AGO), the Organ Consortium at Stanford (ORCAS) and the House Rabbit Society. He holds a B. Sc. in Electrical Engineering and Computing Science from the University of Newcastle upon Tyne, England. E-mail: `ole@interop.com`

Call for Participation

The *Second International Workshop on Community Networking,* subtitled "Integrated Multimedia Services to the Home," will be held June 20–22, 1995 in Princeton, New Jersey, USA. The workshop is sponsored by the IEEE Communications Society in collaboration with ACM SIGCOMM.

Topics

Community networking concerns the network infrastructures that will bring integrated multimedia services to home users. It differs in many ways from enterprise networking in its services, technologies, and economics. In contrast to enterprise networking applications, community networking services will not necessarily be work oriented and will range from entertainment to shopping to information services. At present, community networking technology is driven by the requirements of video-on-demand, most notably high bandwidth (compared to narrowband), bandwidth asymmetry, and the delay-jitter constraints imposed by today's limited-storage TV set-top devices. As various other services develop, community networking will evolve to include integrated multimedia communication and user-to-user applications. Community networking must also provide access to resources located outside the community, in an increasingly global repository of information of every conceivable type. This workshop will give researchers and professionals the chance to share their views and advance the state of the art in this field. Contributions are encouraged in the five areas listed below with relevant topics:

- *Applications and Requirements:* types of applications; coding; set-top operating systems; QoS networking requirements (symmetric/asymmetric bandwidth, delay, and losses); security and privacy; service models; user interface and navigation facilities.

- *Local Distribution Technology:* topology; fiber/cable/UTP/wireless; modulation, bandwidth allocation; MAC (reverse channel); role of ATM; dependencies on equipment/network in the home.

- *Addressing, Signaling, and Upper-Layer Protocols:* local vs. global addressing; the service provider view vs. the common carrier view: the video-dialtone gateway; role of B-ISDN protocols; network- and transport-layer protocols; network management; APIs.

- *Internetworking and Architecture:* the gateway; accessing other networks (data, telephone); server placement and network optimization; the regional distribution centers; testbeds; network traffic models; network cost structure and its implications on service pricing; medium- and long-term network evolution; the impact of regulatory constraints.

- *Community Aspects, Opportunities for Growth:* the success of community networking depends of the degree to which it meets community needs and invites the full participation of community members; community needs, desires and aspirations; networking approaches that have worked well in the past and others that have not; obstacles to success that need to be overcome.

Submissions

Please send via electronic mail a detailed abstract (up to 3 pages in *PostScript* or ASCII) describing a position statement in one of the areas above to: `cn2@arch4.ho.att.com`

Important dates

Deadline for submitting abstracts:	March 17, 1995
Acceptance notification:	April 17, 1995
Papers due (limited to 8 pages):	May 19, 1995

Announcement and Call for Participation

The 9th *USENIX Systems Administration Conference* (LISA IX) will be held September 18–22, 1995 at the Marriott Hotel in Monterey, California. The conferenece is Co-sponsored by USENIX, the UNIX and Advanced Computing Systems Professional and Technical Association, and SAGE, the System Administrators Guild.

The LISA conference is widely recognized as the leading technical conference for system administrators. Historically, LISA stood for "Large Installation Systems Administration," back in the days when having a large installation meant having over 100 users, over 100 systems, or over one gigabyte of disk storage. Today, the scope of the LISA conference includes topics of interest to system administrators from sites of all sizes and kinds. What the conference attendees have in common is an interest in solving problems that cannot be dealt with simply by scaling up well-understood solutions appropriate to a single machine or a small number of workstations on a LAN.

Theme

The theme for this year's conference is "New Challenges," which includes such emerging issues as integration of non-UNIX and proprietary systems and networking technologies, distributed information services, network voice and video teleconferencing, and managing very complex networks. We are particularly interested in technical papers that reflect hands-on experience, describe fully implemented and freely distributable solutions, and advance the state of the art of system administration as an engineering discipline.

Tutorials

The two-day tutorial program on Monday and Tuesday, September 18–19, 1995 offers up to five tracks of full- and half-day tutorials. Tutorials offer expert instruction in areas of interest to system administrators of all levels, from novice through senior. Topics are expected to include networking, advanced system administration tools, Solaris and BSD administration, *Perl* programming, firewalls, NIS, DNS, *Sendmail*, and more. To provide the best possible tutorial offerings, USENIX continually solicits proposals for new tutorials. If you are interested in presenting a tutorial at this or other USENIX conferences, please contact the tutorial coordinator: Daniel V. Klein +1 412 421-0285 FAX: +1 412 421-2332 E-mail: `dvk@usenix.org`

Technical sessions

The three days of technical sessions Wednesday through Friday, September 20–22, 1995 consist of two parallel tracks. The first track is dedicated to presentations of refereed technical papers. The second track is intended to accommodate invited talks, panels and *Works-in-Progress* (WIP) sessions.

Topics

Papers addressing the following topics are particularly timely; papers addressing other technical areas of general interest are equally welcome.

- Dealing with differences in UNIX implementations—migration and interoperability among BSD, SVR4, OSF and others

- Integration of UNIX-based with non-UNIX-based and proprietary systems and networking technologies (Mac, NT and DOS PCs)

- Application of emerging technologies (Mbone, Mosaic) to system administration

- Administration and security of distributed information services (WAIS, Gopher, WWW) and network voice and video teleconferencing (Mbone)

- Experience supporting mobile & location-independent computing

- Experience with large (1000+ machine) networks, especially networks of SVR4-based systems

- Real-world experience with implementations of proposed system administration standards

- Unusual applications of commercial system administration software packages

- Application of operational planning techniques to system administration including measurements and metrics, continuous process improvement, automation, and increasing productivity

- File migration, archival storage and backup systems in extremely large environments

- Innovative tools and techniques that have worked for you

- Managing high-demand and high-availability environments

- Migrating to new hardware and software technologies

- Administration of remote sites that have no technical experts

- Supporting MIS organizations on UNIX

- Real-world experiences with emerging procedural/ethical issues— e.g., developing site policies, tracking abusers, and implementing solutions to security problems

- Networking non-traditional sites (libraries, museums, K–12)

Submission guidelines

An extended abstract is required for the paper selection process. Full papers are not acceptable at this stage; if you send a full paper, you must also include an extended abstract. "Extended" means 2–5 pages. Include references to establish that you are familiar with related work, and, where possible, provide detailed performance data to establish that you have a working implementation or measurement tool.

Submissions will be judged on the quality of the written submission, and whether or not the work advances the state of the art of system administration. For more detailed author instructions and a sample extended abstract, send e-mail to `lisa9authors@usenix.org` or call USENIX at +1 510 528-8649.

Note that the USENIX organization, like most conferences and journals, requires that papers not be submitted simultaneously to more than one conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by "non-disclosure agreement" forms are not acceptable and will be returned unread. All submissions are held in the highest confidence prior to publication in the conference proceedings, both as a matter of policy and as protected by the U.S. Copyright Act of 1976.

Authors of an accepted paper must provide a final paper for publication in the conference proceedings. At least one author of each accepted paper presents the paper at the conference. Final papers are limited to 20 pages, including diagrams, figures and appendixes, and must be in troff, ASCII, or LaTeX format. We will supply you with instructions. Papers should include a brief description of the site, where appropriate.

23

Announcement and Call for Participation *(continued)*

Conference proceedings, containing all refereed papers and materials from the invited talks, will be distributed to attendees and will also be available from the USENIX following the conference.

Where to send submissions

Please submit extended abstracts for the refereed paper track by *two* of the following methods:

- E-mail to: `lisa9papers@usenix.org`

- FAX to: +1 510 548-5738

- Mail to:

  LISA 9 Conference
  USENIX Association
  2560 Ninth Street,
  Suite 215, Berkeley,
  CA 94710
  USA

To discuss potential submissions, and for inquiries regarding the content of the conference program, contact the program co-chairs at `lisa9chair@usenix.org`.

Invited Talk track

If you have a topic of general interest to system administrators, but that is not suited for a traditional technical paper submission, please submit a proposal for a second track presentation to the invited talk coordinators: Laura de Leon, Hewlett-Packard: `deleon@hpl.hp.com` or Peg Schafer, BBN: `peg@bbn.com`

Vendor display

On Wednesday, September 20, well-informed vendor representatives will demonstrate products and services at the informal table-top display. If your company would like to participate, please contact Zanna Knight : `display@usenix.org`

BoFs

Birds-of-a-Feather sessions (BoFs) are very informal gatherings of attendees interested in a particular topic. BoFs are held Tuesday, Wednesday, and Thursday evenings of the conference. BoFs may be scheduled in advance by telephoning the USENIX Conference Office at +1 714 588-8649 or via e-mail to `conference@usenix.org`. They may also be scheduled at the conference.

Registration information

All details of the conference program, conference registration fees and forms, and hotel discount and reservation information will be available in July, 1995. If you wish to receive registration materials, please contact:

USENIX Conference Office
22672 Lambert Street, Suite 613
Lake Forest, CA 92630
USA
Phone: +1 714 588-8649
Fax: +1 714 588-9706
E-mail: `conference@usenix.org`

Important dates

Extended abstracts due:	May 1, 1995
Notification to authors:	June 5, 1995
Final papers due:	August 1, 1995
Registration materials available:	July, 1995

Call for Papers

The Fourth *IEEE International Symposium on High Performance Distributed Computing* (HPDC-4) will be held August 1–4, 1995 at The Ritz Carlton in Pentagon City, Virginia USA. The event is sponsored by IEEE Computer Society TC on Distributed Processing and the Northeast Parallel Architectures Center (NPAC) at Syracuse University, in cooperation with ACM SIGCOMM and Rome Laboratory.

Theme HPDC-4 is a forum for presenting the latest research findings on the application of parallel and distributed computing for solving computationally intensive applications across a network of high-performance computers. Authors are invited to submit full papers on all aspects of high performance distributed computing. Papers that deal with high-level tools, languages, and environments, as well as novel applications, are of particular interest. Papers receiving the best reviews will also be considered for publication in a special issue of the journal *Concurrency: Practice and Experience on High-Performance Distributed Computing*.

Topics Topics of interest include, but are not limited to:

- Software environments and language support for high performance distributed computing

- Parallel and distributed algorithms to solve computationally intensive problems across a LAN, MAN, or WAN.

- High performance I/O and file systems

- Fault tolerance

- Architectural support for high-speed communications or interconnection networks

- Efficient communication interfaces for distributed computing

- Gigabit network architectures

- Networking for multimedia data

- HPDC applications and case studies

Submissions Authors are requested to submit five copies of their manuscript (not to exceed 25 double-spaced pages) to:

Prof. A. S. Grimshaw
Department of Computer Science, Olsson Hall
University of Virginia
Charlottesville, VA 22903–2442
USA
Phone: +1 804-982-2200
E-mail: `grimshaw@virginia.edu`

Important dates

Deadline for paper submissions:	February 3, 1995
Notification of acceptance:	April 25, 1995
Final camera-ready copies:	May 26, 1995

More information For more information about the symposium, send e-mail to:

`hpdc@nova.npac.syr.edu`

EBONE is established as an Association and prepares for 34Mbps

The EBONE Association

In the recent past EBONE, a European Network providing Internet IP and ISO CLNS, has gained substantial stability and its prospects for the future are clear.

EBONE enjoys a rapidly expanding membership, both in terms of capacity and in terms of new members organizations. Seeing this high demand it is no surprise that EBONE is being upgraded continuously. From 1 July 1994 to the end of January 1995, the overall backbone and trans-Atlantic capacity will be increased by 75 % to 12Mbps.

This proves the strong need for EBONE services, therefore the EBONE Consortium of Contributing Organizations decided to re-model EBONE into an Association to provide further stability also in legal terms.

On 2 November 1994 the EBONE Association was formed under French law. Its first Executive Committee officers are:

- Christian Michau from RENATER, France, President,
- Peter Rastl from ACONET, Austria, Vice-President,
- Dave Morton from ECRC, Germany, Treasurer and
- Dennis Jennings from HEANET, Ireland, Secretary.

Technical development

The global Internet is developing at a speed without parallel, with connected organizations multiplying exponentially. A serious challenge to the Internet infrastructure capacity is the so-called multicast services that have recently found a growing market. Therefore EBONE is configuring multicast using *Protocol Independent Multicast* (PIM) in its PoPs in order to reduce multiplication of multicast packets. This new development will be operational by the end of January 1995.
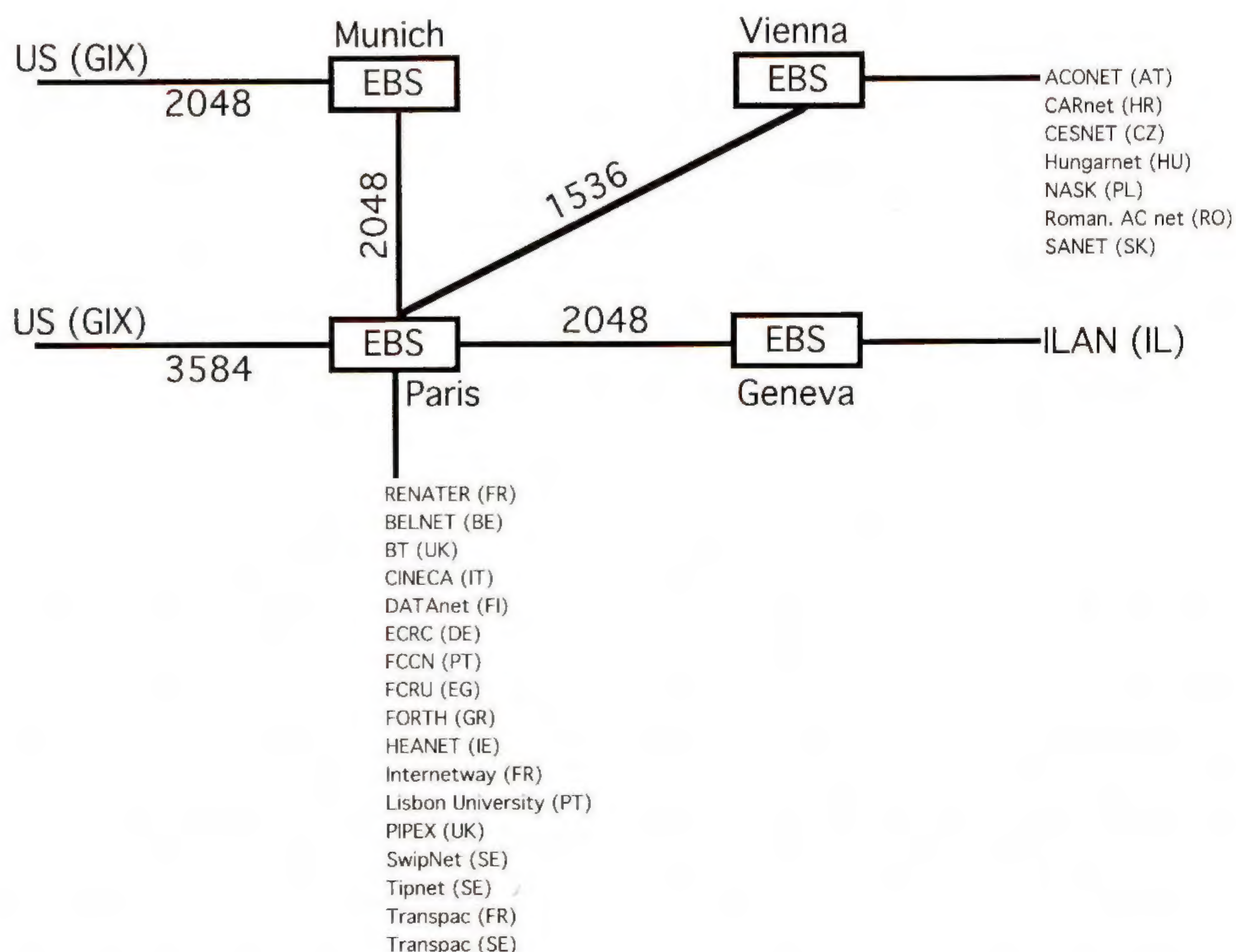

Figure 1: EBONE configuration planned for January 1995

Plans for 1995

The EBONE Association with its state-of-the art-capabilities in terms of equipment, manpower and routing, is in an excellent position to start piloting 34Mbps networking for research in Europe in order to provide high speed services, compliant with the Bangemann report.

EBONE is looking for participation to its initiative to organize a framework for interconnecting between service providers (peering) at 34Mbps in Europe.

EBONE is also interested in participating in the ACTS programme and sees the interconnection of the ACTS's National Hosts as a highly valuable contribution to coordination of national interconnections in Europe. EBONE will put forward a proposal to the EC in the framework of ACTS.

For more information contact:

Christian Michau, EBONE chairman
RENATER, France
Phone: +33 14427 4260,
Fax :   +33 14427 2613
E-mail: `christian.michau@urec.fr`

Frode Greisen, EBONE general manager
UNI-C, Denmark
Phone: +45 3582 8355
Fax: +45 3183 7949
E-mail: `frode.greisen@uni-c.dk`

[Ed.: See also, "EBONE: The European Internet Backbone," by Bernhard Stockman, *ConneXions,* Volume 7, No. 5, May 1993].

Write to *ConneXions*!

We'd love to hear your comments, suggestions and questions about anything you read in *ConneXions*. Our editorial address is given below. Use it for letters to the Editor, questions about back issues etc.:

ConneXions—The Interoperability Report
303 Vintage Park Drive, Suite 201
Foster City, CA 94404–1138, USA
Phone: +1 415-578-6900 or 1-800-INTEROP (Toll-free in the USA)
Fax: +1 415-525-0194
E-mail: `connexions@interop.com`
URL: `http://www.interop.com`

For questions about your subscription please call our customer service hotline: 1-800-575-5717 or +1 502-493-3217 outside the USA. This is the number for our subscription agency, the Cobb Group. Their fax number is +1 502-491-8050. The mailing address for subscription payments is P.O. Box 35840, Louisville, KY 40232–9496.

ADDRESS CORRECTION
REQUESTED

Printed on recycled paper

conneXions

Subscribe to conneXions

U.S./Canada ❏ $150. for 12 issues/year   ❏ $270. for 24 issues/two years   ❏ $360. for 36 issues/three years

International $ 50. additional **per year** (**Please apply to all of the above.**)

Name _____ Title _____

Company _____

Address _____

City _____ State _____ Zip _____

Country _____ Telephone (   ) _____

❏ Check enclosed (in U.S. dollars made payable to **conneXions**).
❏ Visa ❏ MasterCard ❏ American Express ❏ Diners Club Card # _____ Exp.Date _____

Signature _____

Please return this application with payment to: **conneXions**

Back issues available upon request $15./each
Volume discounts available upon request

303 Vintage Park Drive, Suite 201
Foster City, CA 94404-1138
415-578-6900 FAX: 415-525-0194
connexions@interop.com